最大ベイズ境界性の簡易化による高速化について

岸下　昂生†　　片桐　滋†　　大崎美穂†
† 同志社大学理工学部情報システムデザイン学科

1. はじめに

先行研究である最大ベイズ境界性（MBB; Maximum Bayes-Boundaryness）学習法[1]において，分類境界における標本ごとに重みづけられていたが，計算時間削減のために重みを全て 1 として実験を行った．

2. 最大ベイズ境界性学習法

MBB 学習法では以下の 2 ステップを繰り返すことでベイズ境界達成を目指す．

1. ベイズ境界性尺度設定(評価)：ベイズ境界性尺度の推定を学習標本に対して行う．
2. 分類器パラメータ更新(学習)：ステップ 1 で推定したベイズ境界性尺度について，その最大化を学習目標とする分類器パラメータの最適化を行う．

以下では 2 クラス分類について考える．ベイズ境界性尺度をシャノンエントロピー関数により定義する．

$$H_y(\boldsymbol{x}) = -\sum_{j=1}^{2} P\left(C_j \middle| \boldsymbol{x}\right) \log_2 P\left(C_j \middle| \boldsymbol{x}\right) \tag{1}$$

式(1)について，分類事後確率が等しい，つまりいずれのクラスにも分類できない時に最大値 1 に近い値を，一方のクラスに偏る時に最小値 0 に近い値を出力する．

ステップ 2 について，学習に際して損失関数を理想状態と現状の分類器の差として式(2)に定義する．

$$\widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda}) = 1 - \widehat{H}_y(\boldsymbol{x}; \boldsymbol{\Lambda}) \tag{2}$$

ただし，$\boldsymbol{\Lambda}$は分類器パラメータを表す．分類器パラメータの更新には最急降下法を用いる．式(2)は各入力標本$\boldsymbol{x}$に対する損失であるので，全入力標本に対する損失として，経験的平均損失を式(3)に定義する．

$$L(\boldsymbol{\Lambda}) = \sum_{\boldsymbol{x}} W_y(\boldsymbol{x}; \boldsymbol{\Lambda}) \widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda}) \tag{3}$$

ただし，$W_y(\boldsymbol{x}; \boldsymbol{\Lambda})$は最急降下法により，分類器パラメータ$\boldsymbol{\Lambda}$を更新する．ただし，$t$ は学習ステップ数，$\varepsilon > 0$は学習係数をそれぞれ表す．

$$\boldsymbol{\Lambda}^{(t+1)} = \boldsymbol{\Lambda}^{(t)} - \varepsilon \frac{\partial L\left(\boldsymbol{\Lambda}^{(t)}\right)}{\partial \boldsymbol{\Lambda}^{(t)}} \tag{4}$$

式(4)の第 2 項について，積の微分法から

$$\frac{\partial L(\boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} = \sum_{\boldsymbol{x}} \left(\frac{\partial W_y(\boldsymbol{x}; \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda}) + W_y(\boldsymbol{x}; \boldsymbol{\Lambda}) \frac{\partial \widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \right) \tag{5}$$

となる．計算にあたり，コンピュータで数値計算を可能にするため，微分の連鎖律を用いて第 1 項と第 2 項がそれぞれ計算される．

3. MBB 学習法の簡易化

$\widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda})$と$W_y(\boldsymbol{x}; \boldsymbol{\Lambda})\widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda})$はいずれも凸関数であるため，分類器パラメータの更新に用いる最急降下法の勾配にいずれを用いても勾配の正負は一致する．そこで，経験的平均損失を式(6)で再定義する．

$$L(\boldsymbol{\Lambda}) = \sum_{\boldsymbol{x}} \widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda}) \tag{6}$$

それに伴い，式(4)の第 2 項は，

$$\frac{\partial L(\boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} = \sum_{\boldsymbol{x}} \frac{\partial \widehat{U}_y(\boldsymbol{x}; \boldsymbol{\Lambda})}{\partial \boldsymbol{\Lambda}} \tag{7}$$

になるため，時間計算量が削減される．

4. 評価実験

人工データセット(GMM300:2 次元)と実データセット(Abalone01: 7 次元，German: 24 次元)を用いた評価実験を行った（表 1）．また，分類誤り率の参考値として，マルチプロトタイプ（MPT）型分類器に対する CV 法による結果を示す．

表 1　評価実験の結果（実行時間(s)以外の単位：%）

		GMM300	Abalone01	German
既存手法	学習用標本	0.08666664	0.2190476	0.041999996
	試験用標本	0.12914532	0.22930402	0.37800002
	実行時間 (s)	1096	1380	680
提案手法	学習用標本	0.089999974	0.2190476	0.29400003
	試験用標本	0.12393165	0.23076922	0.3
	実行時間 (s)	1078	1208	574
MPT+CV	試験用標本	0.12817	0.22279	0.273

表 1 のように，学習用・標本用データともに分類精度を落とすことなく実行時間を短縮できたことが分かる．また，より高次元の特徴量を持つ Abalone01 や German データセットの短縮率が高くなることが分かった．

5. まとめ

評価実験により提案手法が精度を落とさずに MBB 学習法を高速化することが示された．これ以降，どのような特徴量に対し高速化ができるのか詳しく検証する必要がある．

謝辞：本研究は，科研費 18H03266 の支援を受けて行なった．

参考文献

[1] M. Senda, SPML'19: Proceedings of the 2019 2nd International Conference on Signal Processing and Machine Learning "Maximum Bayes Boundary-Ness Training for Pattern Classification", pp18-28, 2019.

[2] C. Bishop, Pattern Recognition and Machine Learning. New York, Springer Science + Business Media. 2006.